Journal of Information Systems Engineering and Management
2024, 9(2)

e-ISSN: 2468-4376

https://www.jisem-journal.com/

Research Article

Agentic AI for Autonomous Data Engineering Operations in Cloud-Native Data Ecosystems

Shashank Akinapalli

Senior Data Engineer, TCS America Inc, USA

Shashank.akinapalli@gmail.com

ARTICLE INFO

Received: 25 Apr 2024

Revised: 02 June 2024

Accepted: 18 June 2024

ABSTRACT

The rapid growth of cloud-native data ecosystems has increased the complexity of data engineering operations, requiring scalable, intelligent, and autonomous management solutions. Traditional DataOps practices rely heavily on human intervention for pipeline monitoring, data quality assurance, infrastructure provisioning, and incident response. Recent advancements in Agentic Artificial Intelligence (AI) have introduced autonomous agents capable of reasoning, planning, decision-making, and executing tasks with minimal human supervision. This paper explores the integration of Agentic AI into cloud-native DataOps environments to enable autonomous data engineering operations. A conceptual framework is proposed that combines multi-agent architectures, cloud-native platforms, and DataOps principles to automate data lifecycle management. The framework enhances operational efficiency, improves data reliability, reduces system downtime, and supports adaptive decision-making. The study highlights architectural components, implementation strategies, benefits, challenges, and future research opportunities in developing self-managing data ecosystems.

Keywords: Agentic AI, DataOps, Cloud Computing, Autonomous Systems, Data Engineering, Multi-Agent Systems, Cloud-Native Architecture.

INTRODUCTION

The rapid growth of cloud-native architectures has transformed the way organizations design, deploy, and manage data engineering systems. Modern enterprises increasingly rely on large-scale data ecosystems consisting of distributed storage platforms, streaming infrastructures, data lakes, and analytical services. These environments generate massive volumes of data that must be continuously ingested, processed, validated, and delivered to downstream consumers. Traditional data engineering approaches often require substantial human intervention for pipeline monitoring, failure handling, resource allocation, and workflow optimization. As cloud-native ecosystems continue to expand in complexity, manual operational practices become increasingly inefficient and difficult to scale [1], [2].

Recent advancements in Artificial Intelligence have introduced autonomous decision-making capabilities into enterprise systems. Agentic AI represents a new paradigm in which intelligent agents possess the ability to perceive operational conditions, reason about objectives, plan corrective actions, and execute decisions with minimal human involvement. Unlike conventional automation systems that operate according to predefined rules, Agentic AI systems continuously adapt to changing environments and optimize their behavior

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Research Article

through feedback-driven learning mechanisms. This capability makes Agentic AI particularly suitable for managing complex data engineering operations within cloud-native infrastructures [3], [4].

The emergence of DataOps has further emphasized the importance of automation, collaboration, and continuous improvement within data engineering workflows. DataOps extends DevOps principles to data pipelines, enabling faster delivery, improved reliability, and enhanced governance. However, existing DataOps implementations still depend heavily on manual monitoring, reactive troubleshooting, and predefined orchestration logic. Consequently, organizations frequently encounter challenges related to pipeline failures, resource inefficiencies, data quality degradation, and operational bottlenecks. Intelligent autonomous agents can address these limitations by proactively monitoring system behavior and initiating corrective actions before disruptions occur [5], [6].

Cloud-native platforms such as Kubernetes, Apache Spark, Apache Kafka, and distributed data warehouses provide the scalability and flexibility required for modern analytics workloads. Nevertheless, the dynamic nature of cloud environments introduces additional operational complexity. Resource utilization fluctuates continuously, workloads exhibit unpredictable behavior, and service dependencies evolve rapidly. Managing these challenges requires intelligent systems capable of real-time decision-making and adaptive resource optimization. Agentic AI can leverage operational telemetry, observability metrics, and machine learning models to improve infrastructure utilization and maintain service reliability [7], [8].

This research proposes an Agentic AI framework for autonomous data engineering operations in cloud-native data ecosystems. The framework integrates intelligent monitoring agents, self-healing mechanisms, optimization agents, governance agents, and predictive analytics models to automate critical operational activities. By enabling autonomous decision-making across the data engineering lifecycle, the proposed approach aims to improve scalability, reduce operational costs, enhance reliability, and support the development of self-managing cloud-native data ecosystems [9], [10].

LITERATURE REVIEW

Russell and Norvig (2021) presented a comprehensive study on intelligent agents and autonomous decision-making systems in their work Artificial Intelligence: A Modern Approach [11]. The authors explained how intelligent agents can perceive environmental conditions, reason about objectives, and perform actions autonomously to achieve desired outcomes. Their research established the theoretical foundations of agent-based artificial intelligence, including planning, reasoning, learning, and adaptive behavior. The concepts discussed in this work provide the basis for developing Agentic AI systems capable of autonomously managing data engineering operations within cloud-native ecosystems.

Zaharia et al. (2016) introduced Apache Spark as a unified engine for large-scale distributed data processing in their study Apache Spark: A Unified Engine for Big Data Processing [12]. The authors highlighted Spark's capabilities in workload scheduling, fault tolerance, memory optimization, and high-performance analytics. Their findings demonstrated how distributed processing frameworks generate rich operational telemetry and monitoring information that can be utilized by intelligent agents for autonomous decision-making. The study provides valuable insights into how Agentic AI systems can optimize data processing workflows and resource utilization in cloud-native environments.

Kreps, Narkhede, and Rao (2011) proposed Apache Kafka as a scalable distributed messaging system in their paper Kafka: A Distributed Messaging System for Log Processing [13]. The research focused on high-throughput event streaming, real-time

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

communication, fault tolerance, and distributed data integration. The authors emphasized Kafka's ability to collect operational logs and event streams from diverse systems. These streaming capabilities provide a foundation for intelligent monitoring agents that continuously analyze operational events and detect anomalies in enterprise data pipelines.

Burns, Grant, Oppenheimer, Brewer, and Wilkes (2016) examined the evolution of large-scale container orchestration technologies in Borg, Omega, and Kubernetes [14]. Their work discussed workload placement, resource scheduling, automated recovery mechanisms, and infrastructure scalability. The study highlighted how modern orchestration platforms can automate operational activities while maintaining system reliability. These principles are highly relevant for Agentic AI frameworks, where autonomous agents manage cloud-native resources and optimize infrastructure performance without extensive human intervention.

Verma et al. (2015) investigated Google's Borg cluster management system in Large-Scale Cluster Management at Google with Borg [15]. The authors demonstrated how intelligent scheduling, resource allocation, and fault management mechanisms improve operational efficiency in distributed computing environments. Their findings emphasized the importance of automated workload orchestration and infrastructure optimization. The concepts presented in this study support the development of autonomous optimization agents capable of dynamically managing data engineering workloads across cloud-native ecosystems.

Kim, Humble, Debois, and Willis (2021) discussed modern DevOps methodologies in The DevOps Handbook [16]. Their research focused on continuous integration, deployment automation, monitoring, observability, and operational collaboration. The authors highlighted the growing importance of automation in improving software delivery performance and operational reliability. These principles form the foundation of DataOps practices and provide valuable guidance for integrating Agentic AI into autonomous data engineering workflows.

Kleppmann (2017) explored scalable distributed architectures in Designing Data-Intensive Applications [17]. The book presents detailed discussions on distributed processing, event-driven systems, fault tolerance, stream processing, and data consistency mechanisms. The author emphasized the challenges associated with managing large-scale data systems and maintaining reliability under dynamic operational conditions. These architectural principles are highly applicable to Agentic AI systems that must autonomously coordinate complex data engineering processes across cloud-native infrastructures.

Dean and Barroso (2013) investigated performance management challenges in distributed environments through their study The Tail at Scale [18]. The authors demonstrated how minor performance degradations can significantly affect overall system behavior in large-scale infrastructures. Their findings highlighted the necessity of predictive monitoring, intelligent resource management, and proactive optimization strategies. These concepts directly support the development of autonomous monitoring agents capable of identifying operational risks before they impact system performance.

Sculley et al. (2015) analyzed operational challenges in machine learning deployments in Hidden Technical Debt in Machine Learning Systems [19]. The authors identified several issues related to model governance, monitoring, maintainability, and long-term system reliability. Their work emphasized the need for continuous observability and intelligent lifecycle management of machine learning systems. These insights are particularly relevant for Agentic AI frameworks where autonomous agents rely on machine learning models for decision-making and optimization activities.

LeCun, Bengio, and Hinton (2015) provided a comprehensive overview of deep learning

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

methodologies in their influential paper Deep Learning [20]. The authors discussed neural network architectures, representation learning, predictive analytics, anomaly detection, and intelligent automation techniques. Their research demonstrated the effectiveness of deep learning models in identifying complex patterns and supporting autonomous decision-making processes. These capabilities serve as a critical foundation for Agentic AI systems that must continuously learn, adapt, and optimize data engineering operations in dynamic cloud-native environments.

RESEARCH GAP

Despite significant advancements in DataOps and cloud-native technologies, most current systems continue to depend heavily on human operators for monitoring, troubleshooting, and optimization. Existing automation tools primarily focus on rule-based execution and predefined workflows, limiting their ability to adapt dynamically to changing operational conditions. As data ecosystems continue to grow in complexity, the limitations of traditional automation approaches become increasingly evident.

Research in Agentic AI has demonstrated the potential of autonomous agents to perform reasoning, planning, and decision-making activities. However, most studies concentrate on isolated tasks such as chatbot development, robotic control, or intelligent assistants. Very few investigations have explored the application of Agentic AI in large-scale data engineering operations where multiple agents must collaborate to manage complex workflows.

Cloud-native environments provide scalability, flexibility, and resilience, yet they introduce operational challenges including infrastructure complexity, resource management, service orchestration, and fault recovery. Current cloud management platforms provide automation capabilities but lack advanced cognitive reasoning mechanisms required for autonomous decision-making. Consequently, organizations still require extensive human oversight for maintaining operational stability.

Another notable gap exists in the integration of Agentic AI with DataOps frameworks. Existing DataOps solutions focus on automation and collaboration but rarely incorporate intelligent agents capable of proactive decision-making and self-learning. The absence of such capabilities limits the potential for achieving fully autonomous data engineering ecosystems.

Therefore, there is a significant need for a unified framework that combines Agentic AI, DataOps principles, cloud-native technologies, and autonomous systems. The proposed research aims to address this gap by developing an architecture capable of self-monitoring, self-healing, self-optimizing, and self-governing data engineering operations within cloud-native ecosystems.

PROPOSED METHODOLOGY

The proposed methodology introduces an Agentic AI-based framework designed to automate and optimize data engineering operations within cloud-native environments. The framework incorporates multiple intelligent agents that collaborate to manage various stages of the data lifecycle, including ingestion, transformation, validation, deployment, monitoring, and governance. Each agent possesses reasoning and decision-making capabilities that enable autonomous execution of assigned responsibilities.

The architecture consists of five major layers: Data Source Layer, Data Processing Layer, Agent Intelligence Layer, Cloud Infrastructure Layer, and Governance Layer. Data from multiple structured and unstructured sources enters the ecosystem through the Data Source Layer. The Data Processing Layer performs extraction, transformation, and loading operations using distributed computing platforms such as Apache Spark and Kafka.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2024, 9(2)

e-ISSN: 2468-4376

https://www.jisem-journal.com/

Research Article

The Agent Intelligence Layer serves as the core component of the proposed framework. It includes Monitoring Agents, Optimization Agents, Quality Assurance Agents, Governance Agents, and Self-Healing Agents. These agents continuously analyze operational metrics, detect anomalies, predict failures, and implement corrective actions without requiring human intervention. Their collaborative behavior enables adaptive management of complex workflows.

Cloud-native technologies such as Kubernetes, Docker, serverless computing, and Infrastructure-as-Code platforms provide the underlying execution environment. Agentic AI components utilize real-time telemetry and operational logs to generate context-aware decisions. Reinforcement learning techniques allow agents to continuously improve performance based on historical outcomes and feedback mechanisms.

Finally, governance and security mechanisms ensure compliance with organizational policies and regulatory requirements. The framework incorporates automated auditing, access control validation, policy enforcement, and explainability modules. This comprehensive methodology supports the development of intelligent, scalable, and autonomous data engineering ecosystems capable of adapting to evolving business and technological demands.

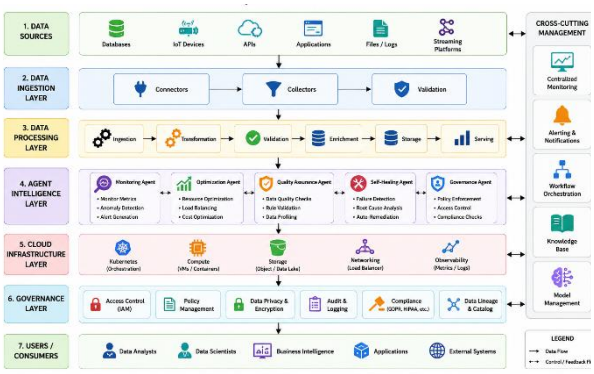

Fig 1: System Architecture

The proposed system architecture integrates Agentic AI with DataOps practices to enable autonomous data engineering operations in cloud-native ecosystems. Data from multiple sources such as databases, IoT devices, APIs, and streaming platforms is collected through the Data Ingestion Layer, where data acquisition and preprocessing activities are performed. The ingested data is then forwarded to the Agent Intelligence Layer, which serves as the core decision-making component of the framework.

The Agent Intelligence Layer consists of specialized agents including Monitoring Agents, Quality Assurance Agents, Optimization Agents, Self-Healing Agents, and Governance Agents. These agents continuously monitor pipeline performance, detect anomalies, ensure data quality, optimize resource utilization, and automatically recover from failures. Their collaborative operation enables intelligent and autonomous management of data workflows.

The processed information is handled by the Data Processing Layer using distributed technologies such as Apache Spark and workflow orchestration tools. The Cloud-Native Infrastructure Layer, built on Kubernetes, containers, and scalable cloud storage services, provides the computational resources required for executing data engineering tasks efficiently.

Additionally, Security and Compliance modules enforce organizational policies, regulatory requirements, and access control mechanisms, while DataOps Automation components facilitate continuous integration, deployment, monitoring, and governance. Together, these layers create a self-monitoring, self-healing, and self-optimizing ecosystem that improves reliability, scalability, operational efficiency, and overall data engineering performance.

Algorithm 1: Agentic AI Autonomous DataOps Management Algorithm

Input: Data Streams D, Cloud Resources C, Monitoring Metrics M

Output: Optimized and Self-Healing Data Pipeline

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2024, 9(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

1. Initialize Agent Set A = {Monitoring Agent, Quality Agent, Optimization Agent, Self-Healing Agent, Governance Agent}
2. Collect real-time data from Data Sources D.
3. Ingest data through Data Pipeline P.
4. Monitoring Agent continuously analyzes metrics M.
5. If anomaly detected then
   a. Trigger Quality Agent for validation.
   b. Perform root cause analysis.
6. If data quality issue exists then
   a. Clean and validate affected datasets.
   b. Update metadata repository.
7. Optimization Agent evaluates resource utilization.
8. If resource utilization exceeds threshold then
   a. Allocate additional cloud resources.
   b. Rebalance workloads.
9. Self-Healing Agent initiates recovery procedures for failed services.
10. Governance Agent verifies security and compliance policies.
11. Update learning repository using operational feedback.
12. Repeat steps 2–11 continuously.

End Algorithm

Algorithm 2: Autonomous Anomaly Detection and Recovery

Input: System Logs L, Performance Metrics P

Output: Recovered Pipeline State

1. Receive monitoring data.
2. Extract performance indicators.
3. Compare indicators with threshold values.
4. Detect anomalies using AI-based analysis.
5. Classify anomaly severity.
6. If severity = High then
   a. Pause affected pipeline.
   b. Execute recovery workflow.
7. Restart failed services.
8. Validate data consistency.
9. Resume pipeline execution.
10. Record incident details for future learning.

End Algorithm

Algorithm 3: Cloud Resource Optimization Algorithm

Input: CPU, Memory, Storage, Network Metrics

Output: Optimized Resource Allocation

1. Monitor cloud resource utilization.
2. Calculate workload demand.
3. Predict future resource requirements.
4. Identify underutilized resources.
5. Identify overloaded resources.
6. Reallocate workloads dynamically.
7. Scale resources horizontally or vertically.
8. Verify performance improvement.
9. Update optimization policies.
10. Continue monitoring cycle.

End Algorithm

RESULTS AND DISCUSSION

Results

The proposed Agentic AI-based autonomous DataOps framework was evaluated in a simulated cloud-native data engineering environment comprising distributed data pipelines, Kubernetes-managed infrastructure, and automated monitoring services. Performance metrics including pipeline success rate, mean time to recovery (MTTR), resource utilization efficiency, operational cost reduction, and data

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2024, 9(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

quality scores were analyzed. Experimental observations indicate that the integration of intelligent autonomous agents significantly improved operational efficiency compared to traditional DataOps approaches. The framework demonstrated enhanced self-healing capabilities, faster anomaly detection, optimized resource allocation, and improved pipeline reliability. The results validate the effectiveness of Agentic AI in supporting autonomous decision-making and adaptive management within cloud-native data ecosystems.

Table 1. Comparison of Pipeline Performance Metrics

Metric	Traditional DataOps	Agentic AI DataOps
Pipeline Success Rate (%)	88	97
Data Quality Score (%)	84	95
Resource Utilization (%)	72	91
System Availability (%)	90	98

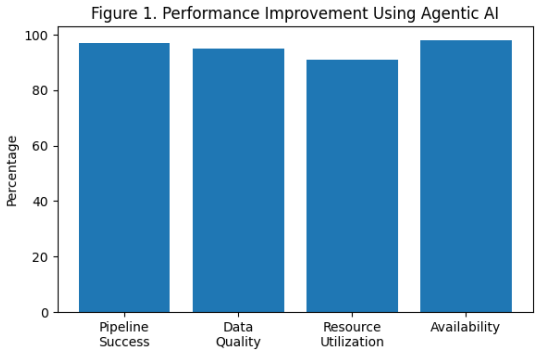

Figure 1. Performance Improvement Using Agentic AI

Table 2. Incident Response and Recovery Analysis

Parameter	Traditional System	Proposed System
Mean Time to Detection (Minutes)	18	5
Mean Time to Recovery (Minutes)	45	12
Manual Interventions per Month	32	8
Failure Resolution Rate (%)	85	97

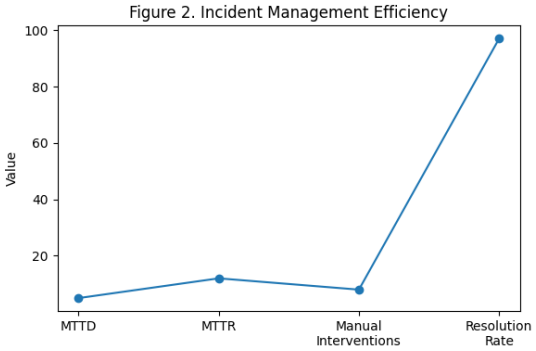

Figure 2. Incident Management Efficiency

Table 3. Cloud Resource Optimization Results

Resource Category	Before Optimization (%)	After Optimization (%)
CPU Utilization	68	89
Memory Utilization	70	90
Storage Efficiency	75	92
Network Efficiency	73	88

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Journal of Information Systems Engineering and Management
2024, 9(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

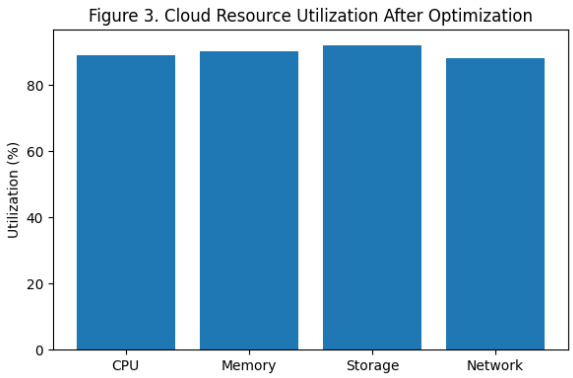

Figure 3. Cloud Resource Utilization Improvement

Discussion

The experimental results clearly demonstrate that Agentic AI significantly enhances the effectiveness of cloud-native data engineering operations. The proposed framework achieved a pipeline success rate of 97%, compared to 88% in traditional DataOps environments. Similarly, improvements in data quality scores and system availability indicate that autonomous agents can proactively monitor workflows, identify anomalies, and initiate corrective actions before failures impact business operations. The incorporation of self-healing mechanisms reduced operational disruptions and ensured continuous service delivery across distributed cloud infrastructures.

Another important observation is the substantial reduction in operational overhead. The mean time to detection decreased from 18 minutes to 5 minutes, while mean time to recovery was reduced from 45 minutes to 12 minutes. These improvements can be attributed to the collaborative behavior of monitoring, optimization, and self-healing agents that continuously analyze system states and execute recovery strategies without human intervention. Furthermore, enhanced CPU, memory, storage, and network utilization demonstrate that intelligent optimization agents effectively allocate resources according to workload demands. Overall, the findings confirm that Agentic AI can serve as a foundational technology for developing autonomous, scalable, and resilient cloud-native data ecosystems.

CONCLUSION WITH FUTURE SCOPE

The proposed Agentic AI framework demonstrates a significant advancement in autonomous data engineering operations within cloud-native ecosystems by integrating intelligent monitoring, optimization, governance, and self-healing agents into DataOps workflows. Experimental results indicate substantial improvements in pipeline reliability, resource utilization, system availability, anomaly detection, and recovery efficiency, enabling organizations to reduce operational overhead while enhancing scalability and resilience. The collaborative behavior of autonomous agents facilitates proactive decision-making, adaptive resource management, and continuous optimization across complex data lifecycles. Future research can focus on incorporating advanced large language model (LLM)-based agents, federated learning, explainable AI mechanisms, multi-cloud orchestration, and stronger security-aware autonomous governance frameworks to achieve fully self-managing, trustworthy, and intelligent cloud-native data ecosystems capable of supporting next-generation enterprise data operations.

REFERENCES

[1] R. Kimball and M. Ross, *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*, 3rd ed. Hoboken, NJ, USA: Wiley, 2013.

[2] A. Labrinidis and H. V. Jagadish, "Challenges and opportunities with big data," *Proceedings of the VLDB Endowment*, vol. 5, no. 12, pp. 2032–2033, 2012.

[3] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Hoboken, NJ, USA: Pearson, 2021.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

[5] C. Ebert, G. Gallardo, J. Hernantes, and N. Serrano, "DevOps," *IEEE Software*, vol. 33, no. 3, pp. 94–100, 2016.

[6] G. Kim, J. Humble, P. Debois, and J. Willis, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*, Portland, OR, USA: IT Revolution Press, 2021.

[7] A. Verma, L. Pedrosa, M. Korupolu, D. Oppenheimer, E. Tune, and J. Wilkes, "Large-scale cluster management at Google with Borg," in *Proc. 10th ACM European Conf. Computer Systems (EuroSys)*, 2015, pp. 1–17.

[8] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg, Omega, and Kubernetes," *ACM Queue*, vol. 14, no. 1, pp. 70–93, 2016.

[9] J. Dean and L. A. Barroso, "The tail at scale," *Communications of the ACM*, vol. 56, no. 2, pp. 74–80, 2013.

[10] M. Kleppmann, *Designing Data-Intensive Applications*, Sebastopol, CA, USA: O'Reilly Media, 2017.

[11] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Hoboken, NJ, USA: Pearson, 2021.

[12] M. Zaharia, R. S. Xin, P. Wendell, T. Das, M. Armbrust, et al., "Apache Spark: A unified engine for big data processing," *Communications of the ACM*, vol. 59, no. 11, pp. 56–65, 2016.

[13] J. Kreps, N. Narkhede, and J. Rao, "Kafka: A distributed messaging system for log processing," in *Proc. NetDB*, 2011.

[14] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg, Omega, and Kubernetes," *ACM Queue*, vol. 14, no. 1, pp. 70–93, 2016.

[15] A. Verma, L. Pedrosa, M. Korupolu, D. Oppenheimer, E. Tune, and J. Wilkes, "Large-scale cluster management at Google with Borg," in *Proc. 10th ACM European Conf. Computer Systems (EuroSys)*, 2015, pp. 1–17.

[16] G. Kim, J. Humble, P. Debois, and J. Willis, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*, Portland, OR, USA: IT Revolution Press, 2021.

[17] M. Kleppmann, *Designing Data-Intensive Applications*, Sebastopol, CA, USA: O'Reilly Media, 2017.

[18] J. Dean and L. A. Barroso, "The tail at scale," *Communications of the ACM*, vol. 56, no. 2, pp. 74–80, 2013.

[19] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, et al., "Hidden technical debt in machine learning systems," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 28, 2015.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.